AA55

by Christian S.k Aditya

Submission date: 06-Jan-2020 01:55PM (UTC+0700)
Submission ID: 1239478167
File name: B6.pdf (776.45K)
Word count: 2965
Character count: 17074

Seminar Nasional Teknologi dan Rekayasa (SENTRA) 2019
ISSN (Cetak) 2527-6042
eISSN (Online) 2527-6050

SISTEM TEMU KEMBALI INFORMASI BUKU HADITS MENGGUNAKAN PEMBOBOTAN TERM FREQUENCY INVERSE DOCUMENT FREQUENCY DAN COSINE SIMILARITY

Christian Sri Kusuma Aditya[1] Vinna Rahmayanti Setyaning Nastiti[2]
[1,2]Universitas Muhammadiyah Malang

Kontak Person:
Christian Sri Kusuma Aditya 081359659715
Jl. Raya Tlogomas 246 Malang 65144
Email: christianskaditya@umm.ac.id

Abstract

Kemajuan dalam bidang teknologi komputer, memungkinkan dapat membantu pekerjaan atau bahkan menggantikan sumber daya manusia yang lebih baik. Sistem Temu Kembali Informasi (STKI) merupakan sebuah sistem yang mempelajari bagaimana menemukan kembali suatu informasi yang sesuai untuk kebutuhan suatu pengguna disekumpulan informasi secara otomatis. Buku Hadits merupakan sebuah buku yang di dalamnya terdiri dari kumpulan perkataan (sabda), percakapan, perbuatan, ketetapan dan persetujuan dari Nabi Muhammad yang dijadikan landasan syariat Islam. Menjadi tidak efisien dan praktis apabila ketika kita ingin mencari informasi pada buku Hadits, harus mencarinya dengan membuka halaman satu-persatu secara manual. STKI dapat dimanfaatkan untuk mencari informasi spesifik, karena dapat memberikkan nilai similarity yang dapat digunakan untuk melakukan pencarian dokumen relevan dengan yang kita inginkan. Pada penelitian ini dilakukan pengujian perbandingan pembobotan TFraw.IDF, TFlog.IDF, dan TFnorm.IDF yang ketiganya merupakan varian dari nilai Term Frequency (TF). Untuk memperkecil term dan mempercepat proses perhitungan term, dilakukan preprocessing meliputi tokenizing, stopword removal atau filtering, dan stemming. Hasil uji coba didapatkan nilai recall rata-rata sebesar 97.3%, dan precision 82.4%.

Keywords: *hadits, TFIDF, cosine similarity, term weighting*

1. Pendahuluan

Hadits adalah salah satu sumber tasyri' dalam Islam. Urgensinya semakin nyata melalui fungsi-fungsi yang dijalankannya sebagai penjelas dan penafsir Al-Qur'an, bahkan sebagai penetap hukum yang independen sebagaimana Al-Qur'an sendiri. Menjadi sangat penting untuk menjaga dan "mengawal" pewarisan ilmu Hadits ini dari generasi ke generasi, misalnya menetapkan berbagai persyaratan yang ketat agar sebuah Hadits dapat diterima (dengan derajat shahih ataupun hasan). Setelah meneliti dan membuktikan keabsahan sebuah hadits secara sanad, tidak cukup berhenti hingga di situ, perlu untuk dikaji matannya hingga mereka dapat menyimpulkan dan mendapatkan Hadits sebagai hujjah.

Kemajuan dalam bidang teknologi komputer, memungkinkan dapat membantu pekerjaan atau bahkan menggantikan sumber daya manusia yang lebih baik. Sistem Temu Kembali Informasi (STKI) merupakan sebuah sistem yang mempelajari bagaimana menemukan kembali suatu informasi yang sesuai untuk kebutuhan suatu pengguna disekumpulan informasi secara otomatis. Oleh sebab hal itu perlu dikembangkannya sebuah sistem teknologi di bidang agama diharapkan dapat mempermudah dan memberikan kenyamanan bagi masyarakat luas untuk mencari dan mengetahui ilmu-ilmu Agama Islam pada sebuah buku Hadits.

Pada penelitiannya tentang STKI yang ditulis oleh Safier Yusuf (2017) memuat tentang pencarian pasal Kitab Undang-Undang Hukum Pidana (KUHP) menggunakan metode *synonym recognition* dan *cosine similarity*. Dalam pengujian penelitian tersebut menggunakan *Query* (Makar, Pemerkosaan, Pencurian, Mengedarkan Mata Uang Palsu, Hukuman mati, Pelanggaran Lalu Lintas, Pembunuhan Berencana, Menjual minuman Keras, Hukuman Seumur hidup, Kejahatan Terhadap Negara) dari seorang pakar Hukum Mendapatkan hasil 52% yang sesuai dengan Pakar sedangkan 48% tidak sesuai [1].

Penelitian lain yang dilakukan oleh Karter (2016) membangun sistem pencarian dengan menghitung kemiripan kumpulan judul dokumen skripsi menggunakan konsep *vector space model* dan

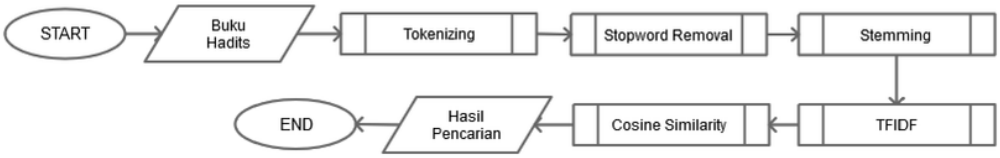
ISSN (Cetak) 2527-6042
eISSN (Online) 2527-6050

metode pembobotan TFIDF Pengujian indexing sebanyak 100 dokumen dengan jumlah *term* sebanyak 9480 *term* yang di berikan bobot mendapatkan nilai Precision 68%. [2]

Berdasarkan uraian di atas, peneliti mengusulkan pengembangan STKI buku Hadits menggunakan pembobotan TFIDF dan metode *cosine similarity*. Metode TFIDF penulis gunakan, karena merupakan metode pembobotan kata yang efisien, mudah dan mampu menjaga nilai *precision*, *recall* dan *f-measure* yang tinggi [3]. Sedangkan metode *cosine similarity* penulis gunakan karena tidak terpengaruh pada panjang pendeknya suatu dokumen. Sehingga, dengan melakukan perbandingan *keyword* yang dihasilkan, maka kedekatan antara *item*-pun dapat dipastikan

2. Metode Penelitian

Pada tahap perancangan sistem, langkah-langkah dalam penyelesaian masalah secara serta perancangan pengujian sistem dapat dilihat pada Gambar 1 yang merupakan diagram alir proses temu kembali informasi secara umum menggunakan pembobotan TFIDF dan pembobotan *cosine similarity*.



Gambar 1 Perancangan Sistem

2.1. Data

Dataset yang didapat adalah dari buku Hadits "Buluughul Maroom min Adillatil Ahkaam" versi 3.0 yang dipublikasikan pada tahun 2010. Didalamnya berisi 16 Kitab seperti Kitab Thoharoh, Kitab Sholat, Kitab Jenazah, Kitab Zakat, Kitab Puasa, Kitab Haji, dan Kitab yang lainnya dan juga terdapat total ada 1.323 potongan Hadits. Dokumen yang diproses pada penelitian ini adalah sudah terjemahan kedalam Bahasa Indonesia.

2.2. Tokenizing

Tahap *tokenizing* atau *parsing* adalah tahap pemotongan kalimat berdasarkan tiap kata yang menyusunnya. Pada tahap tokenizing, karakter spasi digunakan sebagai *delimiter* untuk memecah kalimat menjadi kumpulan kata-kata. Semua tanda baca dan tanda hubung akan dihapuskan, termasuk semua karakter selain huruf alphabet.

2.3. Stopword Removal

Tahap *stopword removal* atau *filtering* adalah tahap mengambil kata - kata penting dari hasil token. *Stopword* adalah kata-kata yang tidak deskriptif yang dapat dibuang dalam pendekatan *bag-of-words*. Contoh *stopwords* dalam Bahasa Indoensia diantaranya adalah "yang", "dan", "di", "dari" dan seterusnya. Data *stopword* yang digunakan diambil dari jurnal Fadillah Z Tala berjudul "*A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia*" [4]. *Stopword* memiliki frekuensi atau kemunculan yang cukup tinggi dan dapat ditemukan hampir dalam setiap dokumen. Oleh karena itu penghilangan *stopword* dapat mengurangi ukuran *index* dan waktu pemrosesan. Selain itu, juga dapat mengurangi level *noise*. Namun terkadang proses *stopword removal* tidak selalu meningkatkan nilai *retrieval*. Pembangunan *stopword* yang kurang hati-hati dapat memperburuk kinerja sistem. Belum ada suatu kesimpulan pasti bahwa penggunaan stopping akan selalu meningkatkan nilai retrieval, karena pada beberapa penelitian, hasil yang didapatkan cenderung bervariasi [5].

ISSN (Cetak) 2527-6042
eISSN (Online) 2527-6050

2.4. Stemming

Pembuatan *term-index* dilakukan karena suatu dokumen tidak dapat dikenali langsung oleh suatu sistem temu kembali informasi atau *Information Retrieval* (IR) sistem. Oleh karena itu, dokumen tersebut terlebih dahulu perlu dipetakan ke dalam suatu representasi dengan menggunakan teks yang berada di dalamnya.

Proses *stemming* adalah proses pemotongan partikel-partikel *term* sehingga menjadi kata dasar dengan mengembalikan semua bentukan kata menjadi bentuk kata dasarnya dengan menghilangkan semua imbuhan (*affixes*) baik yang terdiri dari awalan (*prefixes*), akhiran (*suffixes*) dan kombinasi dari awalan dan akhiran (*confixes*) pada kata urunan.

Teknik *stemming* diperlukan selain untuk memperkecil jumlah indeks yang berbeda dari suatu dokumen, juga untuk melakukan pengelompokan kata-kata lain yang memiliki kata dasar dan arti yang serupa namun memiliki bentuk yang berbeda karena mendapatkan imbuhan yang berbeda. Algoritma *stemming* yang diguanakan adalah Nazief-Adriani. Algoritma Nazief-Adriani menggunakan kamus kata dasar dan mendukung *recoding*, yakni penyusunan kembali kata-kata yang mengalami proses stemming berlebih. [6]
Contoh penerapan algoritma Nazief-Adriani :

berkesering**an** → **berke**sering + **an** (DS) → hapus *Derivation Suffixes* **-an**
ber (DP) + **ke**sering → hapus *Derivation Prefix* **ber-**
ke (DP) + sering → hapus *derivation prefix* **ke-**
sering → *root word*

2.5. TFIDF

Metode TFIDF adalah cara pemberian bobot hubungan suatu *term* terhadap dokumen. TFIDF digunakan untuk mengevaluasi seberapa penting sebuah kata di dalam sebuah dokumen atau dalam sekelompok kata.

Term Frequency (TF) adalah frekuensi dari kemunculan sebuah *term* dalam dokumen yang bersangkutan. Semakin besar jumlah kemunculan suatu *term* dalam dokumen, semakin besar pula bobotnya atau akan memberikan nilai kesesuaian yang semakin besar.

Terdapat beberapa jenis formula yang dapat digunakan, *TF binary* dimana hanya memperhatikan apakah suatu *term* ada atau tidak dalam dokumen, jika ada diberi nilai satu (1), jika tidak diberi nilai nol (0). *TF raw* dimana pembobotan diberikan berdasarkan jumlah kemunculan suatu *term* di dokumen. Contohnya, jika muncul lima (5) kali maka kata tersebut akan bernilai lima (5). Untuk menghindari dominansi dokumen yang mengandung sedikit *term* dalam dokumen, namun mempunyai frekuensi yang tinggi, maka menggunakan formula *TF logarithmic* dengan formula seperti pada persamaan (1). Jika suatu *term* terdapat dalam suatu dokumen sebanyak 5 kali maka diperoleh nilai TF_{ij} 1.699. Tetapi jika *term* tidak terdapat dalam dokumen tersebut, bobotnya adalah nol (0).

$$TF_{ij} = \begin{cases} 1 + log_{10}(f_t, d), & f_{t,d} > 0 \\ 0 & f_{t,d} = 0 \end{cases} \quad (1)$$

TF normalization seperti pada persamaan (2), menggunakan perbandingan antara frekuensi sebuah *term* dengan nilai maksimum dari keseluruhan atau kumpulan frekuensi *term* yang ada pada suatu dokumen.

$$TF_{ij} = a + (1 - a)\frac{ft_d}{ft_{max(d)}} \quad (2)$$

di mana a adalah nilai antara 0 dan 1 dan umumnya ditetapkan ke 0,4, meskipun beberapa penelitian menggunakan nilai 0,5. Notasi a dalam pada persamaan (2) adalah istilah perataan yang perannya meredam kontribusi *term* kedua, yang dapat dilihat sebagai penskalaan TF_{ij} terhadap nilai TF_{ij} terbesar dalam dokumen d.

ISSN (Cetak) 2527-6042
eISSN (Online) 2527-6050

Inverse Document Frequency (IDF) merupakan sebuah perhitungan dimana *term* didistribusikan secara luas pada koleksi dokumen yang bersangkutan. IDF menunjukkan hubungan ketersediaan sebuah *term* dalam seluruh dokumen. Semakin sedikit jumlah dokumen yang mengandung *term* yang dimaksud, maka nilai IDF semakin besar. IDF dihitung dengan menggunakan formula seperti pada persamaan (3).

$$IDF_j = \log\left(\frac{D}{df_j}\right) \quad (3)$$

Dimana D adalah jumlah semua dokumen dalam koleksi sedangkan df_j adalah jumlah dokumen yang mengandung *term* (TF_{ij}). Dengan demikian rumus umum untuk *term weighting* TFIDF adalah penggabungan dari formula TF dengan formula IDF dengan cara mengalikan nilai TF dengan nilai IDF seperti pada persamaan (4).

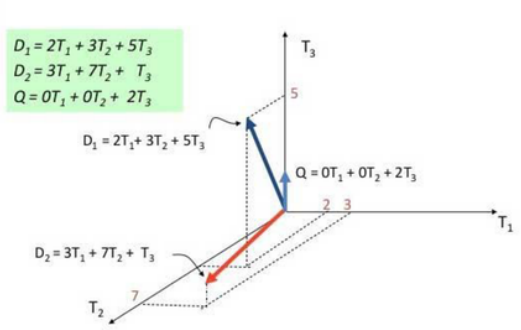
$$w_{ij} = TF_{ij}\, x\, TIDF_j \quad (4)$$

Dimana w_{ij} adalah bobot *term* terhadap dokumen. Berapapun besarnya nilai TF_{ij}, apabila D bernilai sama dengan df_j, maka akan didapatkan hasil 0 (nol) dikarenakan hasil dari log 1, untuk perhitungan IDF. Untuk itu dapat ditambahkan nilai 1 pada sisi IDF, sehingga perhitungan bobotnya menjadi seperti pada persamaan (5).

$$w_{ij} = TF_{ij}\, x \log\left(\frac{D}{df_j}\right) + 1 \quad (5)$$

2.6. Cosine Similarity

Dokumen dalam *Vector Space Model* (VSM) berupa matriks yang berisi bobot seluruh *term* pada tiap dokumen. Bobot tersebut menyatakan kepentingan atau kontribusi *term* terhadap suatu dokumen dan kumpulan dokumen. Kepentingan suatu kata dalam dokumen dapat dilihat dari frekuensi kemunculannya terhadap dokumen. Gambar 2 menunjukkan pemodelan dokumen teks di ruang dimensi dimana D adalah kalimat dokumen sedangkan T adalah *term* atau kata.



Gambar 2 *Vector Space Model*

Misalkan terdapat sejumlah n kata yang berbeda sebagai indeks atau *term-index*. Kata-kata ini akan membentuk ruang vektor yang memiliki dimensi sebesar n. Setiap kata dalam dokumen atau *query* diberikan bobot sebesar w_{ij} seperti pada Gambar 3.

Seminar Nasional Teknologi dan Rekayasa (SENTRA) 2019
ISSN (Cetak) 2527-6042
eISSN (Online) 2527-6050

$$\begin{bmatrix} & T_1 & T_2 & \dots & T_t \\ D_1 & w_{11} & w_{21} & \dots & w_{t1} \\ D_2 & w_{12} & w_{22} & \dots & w_{t2} \\ \dots & \dots & \dots & \dots & \dots \\ D_n & w_{1n} & w_{2n} & \dots & w_{tn} \end{bmatrix}$$

Gambar 3 *Matrix Term*

Cosine similarity digunakan dalam ruang positif, dimana hasilnya dibatasi dengan (0,1) yang mana memberikan tolok ukur seberapa mirip dua dokumen. Dua vektor dengan orientasi besar sudut ruang yang sama, meskipun panjang vektor berbeda akan memiliki nilai kesamaan 1 (satu), atau dapat dikatakan mirip. Persamaan cosine similarity dapat dilihat pada persamaan (6).

$$Cosine(d1, d2) = \frac{d1 \times d2}{|d1| \times |d2|} = \frac{\sum_{i=1}^{n} d1_i \times d2_i}{\sqrt{\sum_{i=1}^{n} d1_i{}^2} \times \sqrt{\sum_{i=1}^{n} d2_i{}^2}} \quad (6)$$

3. Hasil dan Pembahasan

Proses indeks memberikan nilai bobot dari setiap *term* pada dokumen. Beberapa data hasil dari proses indeks dapat dilihat pada Tabel 1. Pada penelitian ini mencoba untuk melakukan komparasi hasil perhitungan dari varian TF_{ij}.

Tabel 1 Hasil Perhitungan *Term Index*

No.	Term	Doc_Id	*TFraw*	*TFlog*	*TFnorm*	*TFraw.IDF*	*TFlog.IDF*	*TFnorm.IDF*
1.	sabda	1	4	1.602	0.431	2.064	0.827	0.222
		20	3	1.477	0.317	1.548	0.397	0.164
		23	2	1.301	0.223	1.032	0.205	0.115
		25	1	1	0.135	0.516	0.516	0.070
2.	wudhu	4	4	1.602	0.721	3.128	1.253	0.564
		12	2	1.301	0.605	1.564	1.017	0.473
		15	3	1.477	0.412	2.346	1.155	0.322
		32	1	1	0.231	0.782	0.782	0.181
3.	hadats	9	1	1	0.259	0.453	0.453	0.117
		12	1	1	0.259	0.453	0.453	0.117
		14	4	1.602	1	1.812	0.726	0.453
		19	2	1.301	0.508	0.906	0.589	0.230
4.	shalat	21	5	1,698	0.769	3.190	1.083	0.491
		25	2	1.301	0.203	1.276	0.830	0.130
		28	1	1	0.121	0.638	0.638	0.077
		67	1	1	0.121	0.638	0.638	0.077
5.	mulia	56	3	1.477	0.892	1.713	0.843	0.509
		64	2	1.301	0.435	1.142	0.743	0.248
		67	2	1.301	0.435	1.142	0.743	0.248
		89	3	1.477	0.892	1.713	0.843	0.509

ISSN (Cetak) 2527-6042
eISSN (Online) 2527-6050

Proses *retrieval* mencari dokumen buku Hadits yang relevan berdasarkan kata kunci atau *query* yang dimasukkan oleh pengguna dan didapatkan nilai *cosine similarity*-nya menggunakan persamaan (6). Hasil dari implementasi proses *retrieval* dapat dilihat pada Tabel 2. Dapat dilihat untuk *query* "bekam bagi orang yang shaum", sistem menemukan 3 buah dokumen yang relevan dengan *similarity* masing-masing 0.431 untuk Doc_Id 23, 0.402 untuk Doc_Id 45 dan 0.349 untuk Doc_Id 67.

Tabel 2 Hasil *Retrieval* oleh Sistem

No.	*Query*	Doc_Id	*Similarity*
1.	bekam bagi orang yang shaum	23	0.431
		45	0.402
		67	0.349
2.	pemakan riba	3	0.345
		52	0.319
		65	0.241
3	sedekah paling mulia	12	0.478
		14	0.459
		19	0.405
		83	0.354
4	menyembelih hewan kurban	34	0.244
		54	0.216

Pengujian sistem untuk proses retrieval akan dilakukan dengan menguji dua parameter utama yaitu *precision* dan *recall*. *Precision* adalah rasio dokumen relevan yang berhasil ditemukembalikan dari seluruh dokumen, dimana nilai tertinggi *precision* adalah 1 yang berarti seluruh dokumen yang ditemukan adalah relevan. *Precision* didefinisikan pada persamaan (7).

$$Precision = \frac{Jumlah\ Dokumen\ Relevan\ Ditemukan}{Jumlah\ Dokumen\ Relevan\ Dalam\ Koleksi} \quad (7)$$

Recall adalah rasio antara dokumen yang relevan yang berhasil ditemukembalikan dari seluruh dokumen relevan yang ada didalam sistem. Nilai tertinggi *recall* adalah 1 yang berarti seluruh dokunen dalam koleksi berhasil ditemukembalikan. *Recall* didefinisikan pada persamaan (8).

$$Recall = \frac{Jumlah\ Dokumen\ Relevan\ Ditemukan}{Jumlah\ Dokumen\ Ditemukan} \quad (8)$$

Hasil pengujian *precision* dan *recall* pada sistem diperlihatkan pada Tabel 3, pengujian dilakukan sebanyak 7 kali menggunakan pembobotan *TFraw.IDF*, dengan menggunakan *query* yang berbeda di setiap pengujian.

Tabel 3 Pengujian *Recall* dan *Precision*

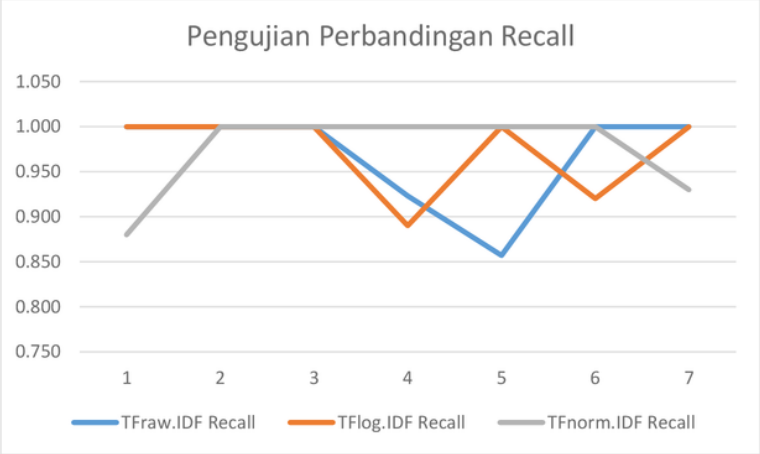
No.	Pengujian	Query	*Ra*	*Rs*	*Rt*	*Recall*	*Precision*
1	T1	bekam bagi orang yang shaum	7	7	10	1	0.7
2	T2	pemakan riba	8	8	15	1	0.533
3	T3	sedekah paling mulia	9	9	13	1	0.692
4	T4	menyembelih hewan kurban	12	13	14	0.923	0.857
5	T5	keutamaan puasa sunah	6	7	10	0.857	0.6
6	T6	amalan yang terus mengalir	11	11	14	1	0.785
7	T7	berbakti kepada orang tua	12	12	13	1	0.923

Dimana notasi *Ra* adalah jumlah dokumen relevan ditemukan, notasi *Rs* adalah jumlah dokumen relevan dalam koleksi, dan notasi *Rt* adalah jumlah dokumen ditemukan.

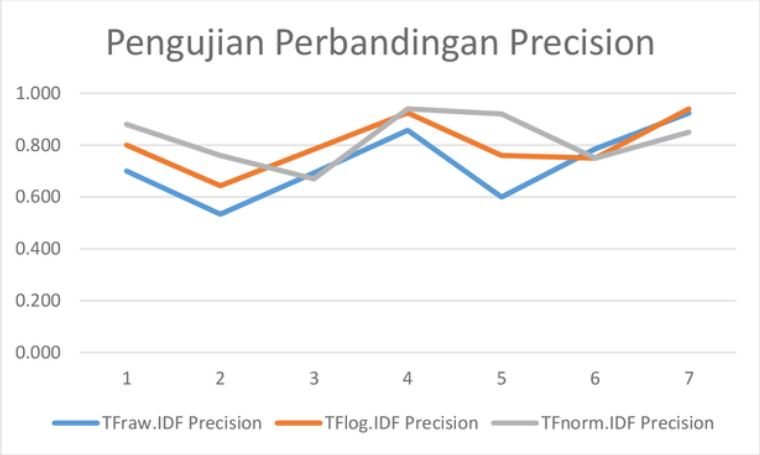
Pada penelitian ini juga membadingankan hasil *precision* dan *recall* dalam penggunaan varian pembobotan *TFraw.IDF*, *TFlog.IDF*, dan *TFnorm.IDF*. Untuk melihat grafik hasil pengujian

ISSN (Cetak) 2527-6042
eISSN (Online) 2527-6050

perbadingan nilai *recall* dapat dilihat pada Gambar 3. Sedangkan grafik hasil pengujian perbadingan nilai *precision* dapat dilihat pada Gambar 4.



Gambar 3. Grafik Perbandingan *Recall*



Gambar 4. Grafik Perbandingan *Precision*

Penggunaan *TFnorm.IDF* pada pengujian *recall* lebih stabil dalam mendapatkan nilai 1 yang artinya seluruh dokumen dalam koleksi yang relevan berhasil di temukembalikan, dibanding penggunaan *TFraw.IDF* maupun *TFlog.IDF*. Nilai dari *precision* memiliki nilai yang berbeda hampir di setiap pengujian, dapat dilihat dari garis grafik yang naik turun. *TFnorm.IDF* mengurangi anomali frekuensi *term* yang lebih tinggi dalam dokumen yang lebih panjang, hanya karena dokumen yang lebih panjang cenderung mengulangi *term* atau kata yang sama berulang-ulang.

Nilai dari *precision* tergantung dari keunikan *query* yang diberikan, semakin unik *query* yang diberikan maka semakin tinggi nilai *precision* yang diperoleh sebaliknya semakin umum *query* yang diberikan maka akan semakin kecil nilai dari *precision* tersebut.

4. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan sistem temu kembali informasi berhasil dibangun dan dapat menemukan dokumen yang relevan pada buku Hadits terhadap *query* yang diberikan pengguna. Pada penelitian ini dilakukan pengujian perbandingan pembobotan *term* menggunakan

ISSN (Cetak) 2527-6042
eISSN (Online) 2527-6050

TFraw.IDF, *TFlog.IDF*, dan *TFnorm.IDF*, didapatkan *TFnorm.IDF* memberikan nilai *recall* lebih stabil mendapatkan nilai 1, nilai maksimal, jika dibanding pembobotan lainnya. Pada *TFnorm.IDF*, semakin panjang *query* tidak terlalu berpengaruh terhadap nilai *recall* karena frekuensi *term* terhadap panjang dokumen akan dinormalisasi, berbeda dengan *TFraw.IDF*, *TFlog.IDF*, dimana semakin panjang query akan berpengaruh terhadap nilai *recall*. Sedangkan pada pengujian *precision* cenderung tidak terlalu memiliki perbedaan, ketiganya akan mendapat nilai *precision* yang lebih optimal pada percobaannya jika menggunakan query yang unik.

Referensi

[1] Yusuf, S., Fauzi, M. A., & Brata, K. C. (2018). Sistem Temu Kembali Informasi Pasal-Pasal KUHP (Kitab Undang-Undang Hukum Pidana) Berbasis Android Menggunakan Metode Synonym Recognition dan Cosine Similarity. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer e-ISSN, 2548, 964X.

[2] Putung, Karter D., Arie SM Lumenta, and Agustinus Jacobus. "Penerapan sistem temu kembali informasi pada kumpulan dokumen skripsi." Jurnal Teknik Informatika 8.1 (2016).

[3] Maarif, Abdul Azis. "Penerapan Algoritma TF-IDF Untuk Pencarian Karya Ilmiah." Teknik Informatika Universitas Dian Nuswantoro, Semarang (2015).

[4] Tala, Fadillah Z. "A study of stemming effects on information retrieval in Bahasa Indonesia." Institute for Logic, Language and Computation, Universiteit van Amsterdam, The Netherlands (2003).

[5] Mahendra, I. P. A. K., and I. Putu Kerta. "Penggunaan Algoritma Semut Dan Confix Stripping Stemmer Untuk Klasifikasi Dokumen Berita Berbahasa Indonesia." Inatitut Teknologi Sepuluh Nopember (2008).

[6] Adriani, Mirna, et al. "Stemming Indonesian: A confix-stripping approach." ACM Transactions on Asian Language Information Processing (TALIP) 6.4 (2007): 1-33.

AA55

ORIGINALITY REPORT

18%	22%	0%	16%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	medium.com Internet Source	5%
2	el-borneo.blogspot.com Internet Source	5%
3	Submitted to UIN Syarif Hidayatullah Jakarta Student Paper	3%
4	Submitted to Universitas Islam Indonesia Student Paper	3%
5	journal.uinjkt.ac.id Internet Source	2%

Exclude quotes	Off	Exclude matches	< 2%
Exclude bibliography	Off		